AMENDMENTS TO THE CLAIMS

1. (Currently Amended) In a graphical modeling and execution environment, a method comprising~~ the steps of~~:

providing a model view and an execution list view of a model being executed, ~~said~~ the model view graphically depicting a plurality of components of ~~said~~ the model, ~~said~~ the execution list view displaying a dynamically updated execution list depicting ~~the~~ an execution order of a plurality of methods called during ~~the~~ an execution of a time step of ~~said~~the model, the dynamically updated execution list changing during the execution of the model to list ~~the plurality of~~ methods that have been called during the time step until a specified point in execution of the time step, ~~said~~ the model view interfaced with a debugger; and

indicating visually a state of the dynamically updated execution list on ~~said~~ the model view at the specified point in the time step.

2. (Currently Amended) The method of claim 1, further comprising~~ the further step of~~:

displaying a visual indicator indicating an association between an executing block method and a calling block on ~~said~~the model view.

3. (Currently Amended) The method of claim 1, further comprising~~ the further step of~~:

displaying a visual indicator indicating an association between a currently executing system method and a subsystem block owner of ~~said~~the currently executing system method on ~~said~~the model view.

4. (Currently Amended) The method of claim 1, further comprising~~ the further steps of~~:

creating a visual representation of a model component not previously displayed in ~~said~~the model view, ~~said~~the model component calling a method; and

displaying a visual indicator indicating an association between the visual representation of the model component not previously displayed and the method called by the model component.

5. (Currently Amended) The method of claim 1, further comprising~~ the further steps of~~:

extending a visual indicator from an originating point to a first called method depicted in ~~said~~the model view; and

extending sequentially ~~said~~the visual indicator to at least one ~~of each~~ subsequently called method depicted in ~~said~~the model view ~~and a virtual subsystem in said model view~~ during a time step in ~~said~~the execution.

6. (Currently Amended) The method of claim 5, further comprising~~ the further step of~~:

indicating ~~the~~ a type of method executing in ~~said~~the model view.

7. (Currently Amended) The method of claim 6 wherein ~~said~~the indication is a visual indication.

8. (Currently Amended) The method of claim 7 wherein ~~said~~the visual indication is made by at least one of one of altering the color of a portion of a model component in ~~said~~the model view representing ~~said~~the method ~~and~~ or inserting a geometric design in a model component displayed in ~~said~~the model view.

9. (Currently Amended) The method of claim 1 wherein a user sets visible breakpoints in ~~said~~the model view.

10. (Currently Amended) The method of claim 9 wherein ~~said~~the breakpoints are conditional breakpoints.

11. (Currently Amended) The method of claim 1, further comprising~~ the further step of~~:

arranging ~~said~~the execution list view to show the methods executed in a current time step in the execution of ~~said~~the model in a tree structure.

12. (Currently Amended) The method of claim 1 wherein a user sets visible breakpoints in ~~said~~the execution list view.

13. (Currently Amended) The method of claim 12 wherein ~~said~~the breakpoints are conditional breakpoints.

14. (Currently Amended) The method of claim 1, further comprising ~~the further step of~~:

setting at least one of a trace point and a display point in at least one of ~~said~~the model view and ~~said~~the execution list view.

15. (Currently Amended) The method of claim 1, further comprising ~~the further steps of~~:

generating at least one of debugging data and profiling data during the execution of ~~said~~the model;

associating ~~said~~the at least one of debugging data and profiling data with at least one of ~~said~~the components of ~~said~~the model; and

visually indicating ~~said~~the associated data in ~~said~~the model view.

16. (Currently Amended) The method of claim 15 wherein ~~said~~the associated data includes solver data.

17. (Currently Amended) The method of claim 1, further comprising ~~the further steps of~~:

generating debugging data with ~~said~~the debugger during the execution of ~~said~~the model;

associating ~~said~~the debugging data with at least one component of ~~said~~the plurality of components of ~~said~~the model; and

visually indicating ~~said~~the associated data in ~~said~~the execution list view.

18. (Currently Amended) The method of claim 17, further comprising ~~the further step of~~:

indicating visually in ~~said~~the execution list view ~~the~~ a number of iterations of at least one ~~of~~ component in ~~said~~the plurality of model components during a time step in ~~said~~the execution.

19. (Currently Amended) The method of claim 1, further comprising ~~the further steps of~~:

selecting a user-set speed parameter via a control associated with ~~said~~the model view; and

executing ~~said~~the model in ~~said~~the model view based on the selected speed parameter.

20. (Currently Amended) The method of claim 1, further comprising ~~the further steps of~~:

selecting a user-set speed parameter via a control associated with ~~said~~the execution list view; and

executing ~~said~~the model in ~~said~~the execution list view based on the selected speed parameter.

21. (Currently Amended) The method of claim 1, further comprising~~ the further steps of~~:

receiving input from a user-controlled input device in ~~said~~the graphical modeling and execution environment, ~~said~~the input being interpreted by ~~said~~the graphical modeling and execution environment as a user-selected speed parameter; and

executing ~~said~~the model in ~~said~~the execution list view based on the selected speed parameter.

22. (Currently Amended) The method of claim 1, further comprising~~ the further steps of~~:

altering at least one of a model component or a connection between ~~said~~the model components~~ and at least one of said model components~~; and

adjusting at least one of ~~said~~the execution list view and ~~said~~the model view to indicate the effects of ~~said~~the altering.

23. (Currently Amended) The method of claim 22 wherein ~~said~~the altering step includes at least one of ~~the~~ adding ~~and~~ or removing of at least one of model components and a connection between ~~said~~the model components.

24. (Currently Amended) The method of claim 1, further comprising~~ the further step of~~:

displaying elements of ~~the~~ a compiled state of ~~said~~the model in ~~said~~the model view.

25. (Currently Amended) The method of claim 1, further comprising~~ the further step of~~:

displaying debug information ~~from said debugger to a user~~ in ~~said~~the model view as a tool tip over a component of ~~said~~the model in response to user input~~.~~.

26. (Currently Amended) The method of claim 25 wherein the displayed debug information indicates a signal value of a signal line in ~~said~~the model view.

27. (Currently Amended) The method of claim 25 wherein the displayed debug information is made persistent in ~~said~~the model view.

28. (Currently Amended) The method of claim 27 wherein ~~said~~the displayed debug information is updated in response to the execution of ~~said~~the model.

29. (Currently Amended) The method of claim 1, further comprising ~~the further step of~~:

displaying debug information ~~from said debugger to a user~~ in ~~said~~the execution list view as a tool tip in response to ~~the~~ a movement of a pointing device in ~~said~~the execution list view over a component of ~~said~~the model associated with ~~said~~the debug information.

30. (Currently Amended) The method of claim 29 wherein the displayed information is made persistent in ~~said~~the execution list view.

31. (Currently Amended) The method of claim 30 wherein ~~said~~the displayed information is updated in response to the execution of ~~said~~the model.

32. (Currently Amended) The method of claim 1, further comprising ~~the further step of~~:

filtering the displayed execution list of methods in ~~said~~the execution list view so that only methods satisfying a user-specified criteria are displayed.

33. (Currently Amended) The method of claim 1, further comprising ~~the further steps of~~:

creating a record for ~~each~~ a unique method invocation; and

displaying data associated with ~~said~~the unique method invocation~~s~~ as the unique method invocation ~~they are~~is called.

34. (Currently Amended) The method of claim 33, further comprising ~~the further step of:~~

anchoring ~~said~~the record to a block owner of ~~said~~the unique method invocation in ~~said~~the model view.

35. (Currently Amended) The method of claim 33, further comprising ~~the further step of~~:

displaying ~~the~~ a calling of ~~said~~the unique method invocation with varying degrees of intensity representative of the frequency of the invocation.

36. (Currently Amended) The method of claim 33, further comprising~~ the further step of~~:

creating a unique method invocation for an execution exception event.

37. (Currently Amended) The method of claim 1 wherein a user sets non-visible breakpoints in at least one of ~~said~~the model view ~~and~~ or ~~said~~the execution list view.

38. (Currently Amended) The method of claim 1 wherein at least one of a set of debugging data ~~and~~ or a set of profiling data are displayed to a user in a separate view.

39. (Currently Amended) A medium holding computer-executable instructions for performing debugging in a graphical modeling and execution environment on an electronic device, ~~said~~the medium comprising:

instructions for providing a model view and an execution list view of a model being executed~~.~~, ~~said~~the model view graphically depicting a plurality of components of ~~said~~the model, ~~said~~the execution list view displaying a dynamically updated execution list depicting ~~the~~ an execution order of a plurality of methods called during ~~the~~ an execution of a time step of ~~said~~the model, the dynamically updated execution list changing during the execution of the model to list ~~the plurality of~~ methods that have been called during the time step until a specified point in execution of the time step, ~~said~~the model view interfaced with a debugger; and

instructions for indicating visually a state of the dynamically updated execution list on ~~said~~the model view at the specified point in the time step.

40. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for displaying a visual indicator indicating an association between an executing block method and a calling block on ~~said~~the model view.

41. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for displaying a visual indicator indicating an association between a currently executing system method and a subsystem block owner of ~~said~~the currently executing system method on ~~said~~the model view.

42. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for extending a visual indicator from an originating point to a first called method depicted in ~~said~~the model view; and

instructions for extending sequentially ~~said~~the visual indicator to ~~each~~ at least one subsequently called method depicted in ~~said~~the model view during a time step in ~~said~~the execution.

43. (Currently Amended) The medium of claim 42, wherein ~~said~~the visual indicator is extended to a virtual subsystem depicted in ~~said~~the model view.

44. (Currently Amended) The medium of claim 42, wherein ~~said~~the medium further comprises:

instructions for indicating ~~the~~ a type of method executing in ~~said~~the model view.

45. (Currently Amended) The medium of claim 44 wherein ~~said~~the indication is a visual indication.

46. (Currently Amended) The medium of claim 45 wherein ~~said~~the visual indication is made by at least one one of altering the color of a portion of a model component in ~~said~~the model view representing ~~said~~the method ~~and~~ or inserting a geometric design in a model component displayed in ~~said~~the model view.

47. (Currently Amended) The medium of claim 39 wherein a user sets visible breakpoints in ~~said~~the model view.

48. (Currently Amended) The medium of claim 47 wherein ~~said~~the breakpoints are conditional breakpoints.

49. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for arranging ~~said~~the execution list view to show the methods executed in a current time step in the execution of ~~said~~the model in a tree structure.

50. (Currently Amended) The medium of claim 39 wherein a user sets visible breakpoints in ~~said~~the execution list view.

51. (Currently Amended) The medium of claim 50 wherein ~~said~~the breakpoints are conditional breakpoints.

52. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for setting at least one of a trace point and a display point in at least one of ~~said~~the model view and ~~said~~the execution list view.

53. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for generating at least one of debugging data ~~and~~ or profiling data with ~~said~~the debugger during the execution of ~~said~~the model;

instructions for associating at least one of ~~said~~the debugging data ~~and~~ or profiling data with at least one of ~~said~~the components of ~~said~~the model; and

instructions for visually indicating ~~said~~the associated data to a user in ~~said~~the model view.

54. (Currently Amended) The medium of claim 53 wherein ~~said~~the associated data includes solver data.

55. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for generating debugging data with ~~said~~the debugger during the execution of ~~said~~the model;

instructions for associating ~~said~~the debugging data with at least one component of ~~said~~the plurality of components of ~~said~~the model; and

instructions for visually indicating ~~said~~the associated data to a user in ~~said~~the execution list view.

56. (Currently Amended) The medium of claim 55, wherein ~~said~~the medium further comprises:

instructions for indicating visually in ~~said~~the execution list view ~~the~~ a number of iterations of at least one component of ~~said~~the plurality of model components during a time step in ~~said~~the execution.

57. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for selecting a user-set speed parameter via a control associated with ~~said~~the model view; and

instructions for executing ~~said~~the model in ~~said~~the model view based on the selected speed parameter.

58. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for selecting a user-set speed parameter via a control associated with ~~said~~the execution list view; and

instructions for executing ~~said~~the model in ~~said~~the execution list view based on the selected speed parameter.

59. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for receiving input from a user-controlled input device in ~~said~~the graphical modeling and execution environment, ~~said~~the input being interpreted by ~~said~~the graphical modeling and execution environment as a user-selected ~~"~~speed parameter~~"~~; and

instructions for executing ~~said~~the model in ~~said~~the execution list view based on the selected speed parameter.

60. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for altering at least one of the model components or a connection between ~~said~~the model components~~ and at least one of said model components~~; and

instructions for adjusting at least one of ~~said~~the execution list view ~~and~~or ~~said~~the model view to indicate the effects of ~~said~~the altering.

61. (Currently Amended) The medium of claim 60, wherein ~~said~~the altering step includes at least one of the adding and removing of at least one of model components ~~and~~ or a connection between ~~said~~the model components.

62. (Currently Amended) The medium of claim 39 wherein ~~said~~the medium further comprises:

instructions for displaying elements of the compiled state of ~~said~~the model in ~~said~~the model view.

63. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for displaying debug information from ~~said~~the debugger to a user in ~~said~~the model view as a tool tip~~-~~ over a component of ~~said~~the model in response to user input.~~-~~

64. (Currently Amended) The medium of claim 63 wherein the displayed debug information indicates a signal value of a signal line in ~~said~~the model view.

65. (Currently Amended) The medium of claim 63 wherein the displayed debug information is made persistent in ~~said~~the model view.

66. (Currently Amended) The medium of claim 65 wherein ~~said~~the displayed debug information is updated in response to the execution of ~~said~~the model.

67. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for displaying debug information from ~~said~~the debugger to a user in ~~said~~the execution list view as a tool tip in response to ~~the~~ a movement of a pointing device in ~~said~~the execution list view over a component of ~~said~~the model associated with ~~said~~the debug information.

68. (Currently Amended) The medium of claim 67 wherein the displayed information is made persistent in ~~said~~the execution list view.

69. (Currently Amended) The medium of claim 68 wherein ~~said~~the displayed information is updated in response to the execution of ~~said~~the model.

70. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for filtering the displayed execution list of methods in ~~said~~the execution list view so that only methods satisfying a user-specified criteria are displayed.

71. (Currently Amended) The medium of claim 39, wherein ~~said~~the medium further comprises:

instructions for creating a record for ~~each~~ a unique method invocation; and

instructions for displaying data associated with one of ~~said~~the unique method invocations as ~~said~~the unique method invocation is called.

72. (Currently Amended) The medium of claim 71, wherein ~~said~~the medium further comprises:

instructions for anchoring ~~said~~the record to a block owner of ~~said~~the unique method invocation in ~~said~~the model view.

73. (Currently Amended) The medium of claim 71, wherein ~~said~~the medium further comprises:

instructions for displaying ~~the~~ a calling of ~~said~~the unique method invocation with varying degrees of intensity representative of ~~the~~ a frequency of the invocation.

74. (Currently Amended) The medium of claim 71, wherein ~~said~~the medium further comprises:

instructions for creating a unique method invocation for an execution exception event.

75. (Currently Amended) The medium of claim 39 wherein a user sets non-visible breakpoints in at least one of ~~said~~the model view ~~and~~or ~~said~~the execution list view.

76. (Currently Amended) The medium of claim 39 wherein at least one of a set of debugging data ~~and~~ or a set of profiling data are displayed to a user in a separate view.

77. (Currently Amended) A system in an electronic device having a graphical design environment, ~~said~~the system comprising:

storage for a debugger, ~~said~~the debugger gathering debug information from the simulation of a model in ~~said~~the graphical design environment; and a display device in communication with ~~said~~the electronic device, the display device displaying:

a model view, ~~said~~the model view displaying a plurality of components of a model and being interfaced with ~~said~~the debugger; and

an execution list view, ~~said~~the execution list view displaying a dynamically updated execution list depicting an execution order of a plurality of methods called during the execution of a time step of ~~said~~the model, the dynamically updated execution list changing during the execution of the model to list ~~the plurality of~~ methods that have been called during the time step until a specified point in execution of the time step, ~~said~~the execution list view ~~state~~ being visually represented on ~~said~~the model view, ~~said~~the execution list view being generated by ~~said~~the debugger.

78. (Currently Amended) The system of claim 77, comprising further:

a visual indicator indicating a currently executing method on ~~said~~the model view.

79. (Currently Amended) The system of claim 78 wherein ~~said~~the visual indicator sequentially extends ~~said~~the indicator to denote ~~said~~the execution order of methods on ~~said~~the model view.

80. (Currently Amended) The system of claim 77 wherein a user is able to set at least one of breakpoints, conditional breakpoints, display points ~~and~~ or trace points on ~~said~~the model view.

81. (Currently Amended) The system of claim 77 wherein a user is able to set at least one of breakpoints, conditional breakpoints, display points ~~and~~ or trace points on ~~said~~the execution list view.

82. (Currently Amended) The system of claim 77 wherein a visual indicator is used to indicate ~~the~~ a type of executing method displayed in ~~said~~the model view.

83. (Currently Amended) The system of claim 82 wherein ~~said~~the visual indicator is one of color ~~and~~ or a geometric pattern.